ESTIMATION OF IMPLICIT USER INFLUENCE FROM PROXY LOGS

An Empirical Study on the Effects of Time Difference and Popularity

Tomonobu Ozaki[1] and Minoru Etho[1,2]

[1]*Cybermedia Center, Osaka University, 1-32 Machikaneyama, Toyonaka, Osaka 560-0043, Japan*

[2]*NTT DOCOMO R&D Center, 3-6 Hikarino-oka, Yokosuka, Kanagawa 239-8536, Japan*

Keywords: User influence, Proxy logs, Web usage mining.

Abstract: In this paper, we propose a framework for estimating implicit user influence from proxy logs. For the estimation, we employ a vector representation of user interactions obtained from log data by taking account of popularity of web pages and difference of access time to them. One of the key issues for successful estimation is how to model the popularity and time difference. Since appropriate models depend on application domains, we propose various models of them. We confirm the effectiveness of the proposed framework by conducting experiments on web page recommendation and community discovery for real proxy logs.

1 INTRODUCTION

Browsing behavior of users on the web is influenced implicitly and explicitly by others. Estimation of the degree of user influence from log data is one of critical tasks for wide variety of applications such as recommendation, viral marketing and community discovery. In this paper, we consider a problem of estimating implicit user influence from proxy logs.

A user modeling from the aspect of *interaction* is required to estimate user influence. We will explain the necessity to model interactions by using a very simple example. In the proxy log shown in Figure 1, while three users `x`, `y` and `z` accessed to the web pages `A.html` and `B.html` in common, we can guess that the degree of influence among them is not equal. While `y` always accesses the same web pages just after `x`'s accesses, the access time of `z` is completely different from those of `x` and `y`. Thus, we can easily expect that the behavior of `x` gives significant impact on that of `y`, and the degree of influence of `x` on `y` is high. Besides the difference of access time, popularity of web page is a promising indicator of user interaction. Since all users except `z` accessed to `A.html` in a short period of time, we can judge that their browsing behaviors on `A.html` might be caused by not user influence but by global one. This very simple example shows that taking account of page popularity and time difference is one of key issues for accurate modeling of user interaction and for estimation of user influence.

UID	URL	Time
x	http://xxx/A.html	2011-04-01 10:01:40
y	http://xxx/A.html	2011-04-01 10:02:21
z	http://xxx/B.html	2011-04-01 10:02:48
m	http://xxx/A.html	2011-04-01 10:08:06
n	http://xxx/A.html	2011-04-01 10:10:15
...	...	...
x	http://xxx/B.html	2011-04-01 15:12:59
y	http://xxx/B.html	2011-04-01 15:14:01
...	...	...
z	http://xxx/A.html	2011-04-01 20:09:10
...	...	...

Figure 1: An example of proxy log.

In this paper, we propose a model of user interactions based on the page popularity and time difference, and develop methods for estimating implicit user influence. In the area of social network analysis, many sophisticated methods for estimating user influence have been proposed, most of which assume link formation representing user interactions. However, we cannot always expect precise link information in case of proxy logs. So, we prepare two methods for the estimation: one does not require link information, and the other works with additional (incomplete) information.

While we focus on the user influence in this paper, the property of *homophily*(McPherson et al., 2001)

Ozaki T. and Etho M..
ESTIMATION OF IMPLICIT USER INFLUENCE FROM PROXY LOGS - An Empirical Study on the Effects of Time Difference and Popularity.
DOI: 10.5220/0003659702420247
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* (KDIR-2011), pages 242-247
ISBN: 978-989-8425-79-9

will also give significant impact on user behavior. Homophily is the tendency of users to have similar behaviors with ones having similar characteristics. In this paper, we drive a rough effect of homophily from log data by using a simple model, and compare it with the effect of influence. In addition, we consider the mixture of homophily and influence.

The effectiveness of the proposed framework is evaluated empirically by conducting experiments on web page recommendation and community discovery.

2 MODELING THE DEGREE OF INFLUENCE

A proxy log $\mathcal{L}$ consists of a set of triplets $l = (u, p, t)$ which indicates that a user u visited or accessed a web page p at time t. We use notations $\mathcal{U}_{\mathcal{L}} = \{u|(u,p,t) \in \mathcal{L}\}$ and $\mathcal{P}_{\mathcal{L}} = \{p|(u,p,t) \in \mathcal{L}\}$ to denote a set of all users and web pages in $\mathcal{L}$, respectively.

Our purpose in this paper is to estimate the degree of influence from a user x to other user y for every ordered pair $\langle x,y\rangle \in \mathcal{U}_{\mathcal{L}} \times \mathcal{U}_{\mathcal{L}}$ of users in $\mathcal{L}$.

2.1 Representation of Interactions

For an ordered pair $\langle x,y\rangle$ of users, we employ an *interaction vector* to represent interactions from x to y on each web page p (see Figure 2). The value of dimension p in an interaction vector is denoted as $V_x^y(p)$.

user pair	p_1	$\cdots$	$p_{\lvert\mathcal{P}_L\rvert}$
$\langle u_1, u_2\rangle$	$V_{u_1}^{u_2}(p_1)$	$\cdots$	$V_{u_1}^{u_2}(p_{\lvert\mathcal{P}_L\rvert})$
$\cdots$	$\cdots$	$\cdots$	$\cdots$
$\langle u_1, u_{\lvert\mathcal{U}_L\rvert}\rangle$	$V_{u_1}^{u_{\lvert\mathcal{U}_L\rvert}}(p_1)$	$\cdots$	$V_{u_1}^{u_{\lvert\mathcal{U}_L\rvert}}(p_{\lvert\mathcal{P}_L\rvert})$
$\cdots$	$\cdots$	$\cdots$	$\cdots$
$\langle u_{\lvert\mathcal{U}_L\rvert-1}, u_{\lvert\mathcal{U}_L\rvert}\rangle$	$V_{u_{\lvert\mathcal{U}_L\rvert-1}}^{u_{\lvert\mathcal{U}_L\rvert}}(p_1)$	$\cdots$	$V_{u_{\lvert\mathcal{U}_L\rvert-1}}^{u_{\lvert\mathcal{U}_L\rvert}}(p_{\lvert\mathcal{P}_L\rvert})$

Figure 2: Vector representation of interactions.

To make $V_x^y(p)$ reflect significance of interaction, we formulate $V_x^y(p)$ in the exponential waiting time model(Gomez Rodriguez et al., 2010) with the consideration of importance of p. In the formulation, we give high value to $V_x^y(p)$ if p is important and y's access time to p is close to that of x. In other words, we regard that x affects y significantly if y follows x's behavior on important web pages. The formal definition is given below:

$$V_x^y(p) = \begin{cases} I_x^y(p) \cdot \exp(-\Delta_x^y(p)/\alpha) & \\ \qquad (\min_{(x,p,t_x)\in\mathcal{L}}(t_x) < \min_{(y,p,t_y)\in\mathcal{L}}(t_y)) & \\ 0 \qquad (otherwise) & \end{cases}$$

where α is a parameter, $I_x^y(p)$ denotes an importance of p with respect to $\langle x,y\rangle$, and $\Delta_x^y(p)$ denotes a difference of timestamps when x and y visited p.

Various models of $I_x^y(p)$ and $\Delta_x^y(p)$ in $V_x^y(p)$ can be considered. In this paper, we examine four models of $I_x^y(p)$ and two of $\Delta_x^y(p)$.

The first model of $I_x^y(p)$ is the inverse document frequency (IDF) of p, defined formally as:

$$\mathrm{idf}(p) = \log\left(\frac{|\mathcal{U}_{\mathcal{L}}|}{|\{z|(z,p,t') \in \mathcal{L}\}|}\right).$$

In this setting, web pages accessed by fewer users have higher importance.

The second model of $I_x^y(p)$ is restricted version of IDF. Only triplets before y's first access to p are used in calculating IDF.

$$\mathrm{r\_idf}(y,p) = \log\left(\frac{|\mathcal{U}_{\mathcal{L}}|}{|\{z|(z,p,t') \in \mathcal{L}, t' \leq \min_{(y,p,t)\in\mathcal{L}}(t)\}|}\right)$$

r_idf(y,p) reflects a context on p and y by considering the access time of y to p. It gives high value to early adopters of p.

As the third model of $I_x^y(p)$, we consider the term frequency - inverse document frequency (tf-idf) defined below. In this case, $I_x^y(p)$ depends on x and p.

$$\mathrm{tfidf}(x,p) = \frac{|\{(x,p,t) \in \mathcal{L}\}|}{|\{(x,p',t') \in \mathcal{L}\}|} \times idf(p)$$

Finally, as the fourth model, we prepare a constant function, *i.e.* $I_x^y(p) = 1$.

Capturing the time difference on a web page p between two users x and y is not trivial since users visit the same web pages several times. To reflect a situation in which y visits p by the influence of x, it is reasonable to use the y's first access to p and x's access just before y's first access. On the other hand, if we assume that x's interest in p decreases with time and thus x's effect on p also decreases, using the first accesses of y and x is another reasonable candidate. To model the above ideas, two models of time difference, denoted as $LtoF_x^y(p)$ and $FtoF_x^y(p)$, are defined:

$$LtoF_x^y(p) = \min_{(y,p,t_y)\in\mathcal{L}}(t_y) - \max_{(x,p,t_x)\in\mathcal{L}_y^p}(t_x)$$
$$FtoF_x^y(p) = \min_{(y,p,t_y)\in\mathcal{L}}(t_y) - \min_{(x,p,t_x)\in\mathcal{L}}(t_x)$$

where $\mathcal{L}_y^p = \{(z,p,t_z) \in \mathcal{L} \,|\, t_z < \min_{(y,p,t_y)\in\mathcal{L}}(t_y)\}$ represents a set of triplets in $\mathcal{L}$ whose time stamp is earlier than y's first access to p.

2.2 Estimation of User Influence

For every ordered pair $\langle x,y\rangle$ of users, an interaction vector can be obtained by instantiating $I_x^y(p)$ and

$\Delta_x^y(p)$ for all web pages $p \in \mathcal{P}_L$. Then, the vectors will be used to estimate a user influence. In this paper, we propose two methods for estimating user influence from a set of interaction vectors.

The first method is very simple. We estimate the degree of influence from x to y, denoted as $w_\sigma(x,y)$, as the *summation of elements* in a vector on $\langle x,y\rangle$:

$$w_\sigma(x,y) = \sum_{p\in\mathcal{P}_L} V_x^y(p).$$

In addition, if necessary, we use a normalized influence $w'_\sigma(x,y) = w_\sigma(x,y)/max_{z\in\mathcal{U}_L}(w_\sigma(z,y))$. As explained before, $V_x^y(p)$ indicates the degree of significance on the interaction from x to y on p. Thus, the estimation by summation gives high degree of influence to $\langle x,y\rangle$ if there are many significant interactions between two users. The idea behind this estimation is related to the traditional similarity measures which give high similarity to the pair of vectors having many high value elements in common. In case of $w_\sigma(x,y)$, the information on "high value elements in common" between x and y is already encoded in calculating $V_x^y(p)$ since $V_x^y(p)$ reflects the significance of *interactions*.

The second proposed method to estimate user influences is application of *supervised learning*. While it is difficult to observe interactions and influences directly in general, we prepare a class information $c : \mathcal{U}_L \times \mathcal{U}_L \to \{0,1\}$ by using additional information which indicates whether or not a user pair has a lot of chances of interactions: $c(x,y) = 1$ means that there is a high possibility of interaction and thus we regard that x influences y significantly, while $c(x,y) = 0$ corresponds to the opposite situation.

A model which estimates the probability that $c(x,y) = 1$ can be obtained by applying a supervised learning to a set of interaction vectors with class information, *i.e.*

$$\{(\langle V_x^y(p_1),\cdots,V_x^y(p_{|\mathcal{P}_L|})\rangle,\ c(x,y)) \,|\, x,y \in \mathcal{U}_L\}.$$

We regard this probability as the degree of influence from x to y and denote it as $w_L(x,y)$. Similar to the case of w_σ, we use the normalized influence $w'_L(x,y) = w_L(x,y)/max_{z\in\mathcal{U}_L}(w_L(z,y))$ if necessary.

The property of homophily(McPherson et al., 2001) also gives significant impact on user behavior. In this paper, we regard that the cosine similarity of user behavior

$$w_C(x,y) = \frac{\sum_{p\in\mathcal{P}_L} \mathrm{tfidf}(x,p)\cdot\mathrm{tfidf}(y,p)}{\sqrt{\sum_{p\in\mathcal{P}_L}\mathrm{tfidf}(x,p)^2}\sqrt{\sum_{p\in\mathcal{P}_L}\mathrm{tfidf}(y,p)^2}}$$

roughly represents homophily effects and use it as a baseline method. In addition, we consider a mixture of homophily and influence:

$$w_I^\lambda(x,y) = \lambda\frac{w_C(x,y)}{max_{z\in\mathcal{U}_L}(w_C(z,y))} + (1-\lambda)w'_I(x,y)$$

where λ is a mixture parameter and $I \in \{\sigma, L\}$.

3 EXPERIMENTS

The proposed framework is evaluated by tasks of web page recommendation and community discovery.

3.1 Datasets

After the application of standard data cleaning, three datasets L_1, L_2 and L_3 are prepared from a proxy server log recorded in Osaka University from April to June 2010. In addition, as a simple abstraction for better estimation, all parameters in URL (string after "?") are deleted.

L_1: It contains about 308,000 records of 99 students who belong to a certain department on sciences.

L_2: It contains about 258,000 records of 151 students who belong to a certain department on arts.

L_3: It contains about 242,000 records of 157 students participating in a certain project.

We prepare class information for L_1 and L_2 based on the physical location of computers determined by IP address recorded in the original proxy log. We judge $c(x,y) = 1$ if there exists at least one situation in which two students x and y use two computers located adjacent to each other at the same time. As a result, the numbers of user pairs $\langle x,y\rangle$ judged as $c(x,y) = 1$ become 786 in L_1 and 776 in L_2, respectively. We prepare class information for L_3 by using 'group information' obtained by a questionnaire. The students in L_3 consists of six groups having 50, 50, 26, 13, 10, and 8 members, respectively. We judge $c(x,y) = 1$ if x and y belong to the same group.

3.2 Web Page Recommendation

3.2.1 Estimation of User Influence

We prepare six settings on α for the exponential waiting time model, denotes as D_5, D_{10}, D_{20}, H_{75}, H_{150} and H_{300}, respectively. In case of D_a ($a = \{5,10,20\}$), we abstract timestamps at the level of "day" and set the parameter α to a. On the other hand, H_a ($a = \{75,150,300\}$) denotes the abstraction of timestamps at the level of "hour". While D_{10} corresponds to the situation in which the effect of page importance decreases to about 0.5 in a week, H_{150} cuts down the effect to about 0.3 in the same period.

By considering all the combinations of $I_x^y(p)$, $\Delta_x^y(p)$ and α, 48($= 4\times 2\times 6$) sets of interaction vectors are obtained for each datasets. From each set

Table 1: Number of records for web page recommendation.

	$\|P_{i,1}\|$	$\|A_{i,1}\|$	$\|P_{i,2}\|$	$\|A_{i,2}\|$
$i=1$	38,827	140,421	104,221	88,303
$i=2$	25,347	35,079	21,367	38,663
$i=3$	25,962	61,432	30,171	67,276

of interaction vectors, we derive w_σ by summation and w_L by supervised learning of LibSVM(Chang and Lin, 2001). Parameters for SVM learning were determined by the grid search. We employ w_C as a baseline. The mixtures w_σ^λ and w_L^λ are also obtained by setting $\lambda = 0, 0.05, 0.1, \cdots, 0.95, 1$, respectively.

3.2.2 Evaluation Metrics

For each $L_i\,(i = \{1,2,3\})$, two pairs of datasets $L_{i,j} = (P_{i,j}, A_{i,j})\,(j = \{1,2\})$ are prepared from the same proxy server log recorded in July 2010. While $P_{i,j}$ is a set of records of students in L_i for one week, $A_{i,j}$ is a set of records for two weeks just after $P_{i,j}$. $P_{i,j}$ and $A_{i,j}$ are used for producing a recommendation set and an answer set, respectively. Different from L_i, we do not apply the abstraction of URL to $L_{i,j}$. The numbers of records are summarized in Table 1.

For each user x, a set of web pages to which x does not access in $P_{i,j}$ is produced as a recommendation set $P_{i,j}(x) = \{p|(z,p,t) \in P_{i,j}, z \neq x\} \setminus \{p|(x,p,t) \in P_{i,j}\}$. Each web page p in the recommendation set has the score $v(p,x) = \sum_{z \in \{z \neq x\,|\,(z,p,t) \in P_{i,j}\}} w(z,x)$ of weighted voting according to a user influence w. We sort $P_{i,j}(x)$ in descending order of the scores. On the other hand, we define the answer set as

$$A_{i,j}(x) = \{p|(x,p,t) \in A_{i,j}, (x,p,t') \notin P_{i,j}\} \cap P_{i,j}(x).$$

We believe that recommendation of minor web pages is worth more than that of major ones. To reflect such consideration, we put a weight $w(p)$ on a web page p based on inverse document frequency, *i.e.*

$$w(p) = \log\left(|\mathcal{U}_{P_{i,j}}| / |\{z|(z,p,t') \in P_{i,j}\}|\right).$$

We employ the macro average of weighted precision@k taken over users as an evaluation criterion. The weighted precision@k for a user x is defined as :

$$p@k(x) = \sum_{p \in P_{i,j}(x)} I(x,p,k) \cdot w(p) \,/ \sum_{p \in P_{i,j}(x)} w(p)$$

where $I(x,p,k)$ is an indicator function which becomes 1 if p is in $A_{i,j}(x)$ and it also locates within the k-th place in $P_{i,j}(x)$. Otherwise, $I(x,p,k) = 0$.

As another evaluation criterion, mean average precision (MAP) is employed:

$$\text{MAP} = \frac{1}{|\mathcal{U}_{A_{i,j}}|} \sum_{x \in \mathcal{U}_{A_{i,j}}} \frac{1}{|A_{i,j}|} \sum_{p \in A_{i,j}(x)} p@k(x,p)(x)$$

where $k(x,p)$ is the rank of p in $P_{i,j}(x)$.

3.2.3 Results

Table 2 shows the best values of MAP among all the combinations of parameters. The best values within each $L_{i,j}$ are marked by underline. We can observe that the proposed methods outperform the baseline (w_C). In addition, the mixtures of homophily and influence (w_σ^λ and w_L^λ) take the first place in all cases. In comparison with the results by summation (w_σ and w_σ^λ), results by supervised learning (w_L and w_L^λ) are better in all cases of $L_{3,j}$. On the other hand, such tendency is not recognized in $L_{1,j}$ and $L_{2,j}$.

Table 2: Best values of MAP.

MAP	$L_{1,1}$	$L_{1,2}$	$L_{2,1}$	$L_{2,2}$	$L_{3,1}$	$L_{3,2}$
w_C	0.231	0.162	0.167	0.269	0.210	0.293
w_σ	0.250	<u>0.198</u>	0.191	0.299	0.240	0.308
w_L	0.253	0.191	0.166	0.303	0.242	0.311
w_σ^λ	<u>0.260</u>	<u>0.198</u>	<u>0.194</u>	0.306	0.243	0.321
w_L^λ	<u>0.260</u>	0.194	0.170	<u>0.310</u>	<u>0.253</u>	<u>0.330</u>

We show the average values of MAP and precision@k ($k = \{5,10\}$) for w_σ and w_L taken over 48 combinations of parameters in Table 3. In the table, all average MAP values except w_L for $L_{2,1}$ and w_σ for $L_{3,2}$ outperform those of baseline method. Similar to MAP, average values of precision@k tend to be higher than corresponding values of baseline method. While w_L is clearly better than w_σ in $L_{2,2}$, $L_{3,1}$ and $L_{3,2}$, w_L is worse in others, especially in $L_{2,1}$.

From the results, we simply conclude that: (1)the proposed methods perform well under appropriate parameter settings, (2)the mixture of homophily and influence gains the result of recommendation, and (3)the quality of class information has an impact on user influence obtained by supervised learning.

Table 3: Average values of MAP and precision@k.

MAP	$L_{1,1}$	$L_{1,2}$	$L_{2,1}$	$L_{2,2}$	$L_{3,1}$	$L_{3,2}$
w_C	0.231	0.162	0.167	0.269	0.210	0.293
w_σ	0.241	0.180	0.173	0.281	0.218	0.287
w_L	0.245	0.188	0.157	0.299	0.235	0.305
precision@5						
w_C	0.436	0.337	0.152	0.343	0.310	0.387
w_σ	0.440	0.369	0.162	0.348	0.334	0.385
w_L	0.422	0.358	0.111	0.357	0.342	0.410
precision@10						
w_C	0.310	0.230	0.134	0.219	0.246	0.310
w_σ	0.347	0.251	0.150	0.239	0.262	0.307
w_L	0.348	0.233	0.124	0.250	0.271	0.339

In order to assess the effects of parameters, we compare the MAP values in all datasets obtained by different models of page importance $I_x^y(p)$ under the

same settings other than $I_x^y(p)$. For each proposed methods w_σ and w_L, we have 72 comparisons in total because of two of time differences, six of αs and six of datasets. The ratio of taking the best value is summarized in Table 4. We apply the same comparisons to $\Delta_x^y(p)$ and α. The results on $\Delta_x^y(p)$ and α are obtained from 144 and 48 comparisons, respectively. While tfidf and const drive better results in w_σ, r_idf is the best in w_L. H_{300} clearly outperforms others in w_σ. *LtoF* is better than *FtoF* in both w_σ and w_L. Since the winning rates are not uniform, we can recognize that different models give significant impact on the results.

Table 4: Winning rates of different models in MAP.

	w_σ	w_L		w_σ	w_L
idf	0.153	0.139	D_5	0.000	0.083
r_idf	0.125	0.347	D_{10}	0.063	0.083
tfidf	0.347	0.181	D_{20}	0.313	0.250
const	0.375	0.333	H_{75}	0.104	0.229
FtoF	0.222	0.333	H_{150}	0.104	0.146
LtoF	0.778	0.667	H_{300}	0.417	0.208

A similar analysis is also applied to a mixture parameter λ in w_σ^λ and w_L^λ. The results are shown in Figure 3. The value of λ between 0.5 and 0.55 and that between 0.65 and 0.75 seem to be promising for w_σ^λ and w_L^λ, respectively. Compared with w_σ^λ, the peak of w_L^λ exists at the higher value of λ. In other words, w_L^λ requires large effect of homophily to get better results on web page recommendation. We believe that unreliability of class information of L_1 and L_2 causes these results.

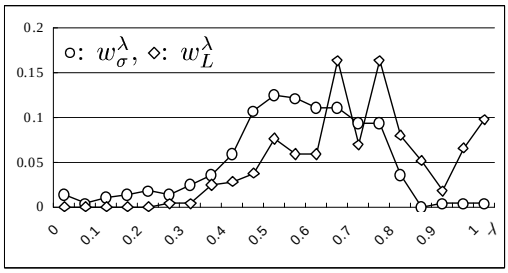

Figure 3: Winning rates of different λs in MAP.

3.3 Community Discovery

We conduct experiments on community discovery by using the dataset L_3.

As the same as the experiments on web page recommendation, we prepare 48 of w_σs for each combination of parameters. On the other hand, we employ a cross-validation like method to derive w_L. In the method, a set of interaction vectors is divided into five pieces and the influence $w_L(x,y)$ in one piece is estimate by using a model build from other four pieces.

A community structure having maximal modularity(Newman and Girvan, 2004) is discovered by using the igraph library(Csardi and Nepusz, 2006). By using the group information obtained by a questionnaire as a correct answer, we evaluate the discovered community structure based on normalized mutual information (NMI)(Danon et al., 2005). The range of NMI is from 0 to 1, and high value indicates that the predicted structure is similar to the answer.

The best and average values of NMI over all the combinations of parameters are shown in Table 5. In the results, we observe that the proposed methods outperform the baseline method w_C. Especially, the best value of w_L is significant. But it is not surprising since we use class information to prepare w_L even if a cross-validation like method is applied. Different from the results in web page recommendation, the mixtures w_σ^λ and w_L^λ become worse than w_σ and w_L. We believe that the normalization process causes these results. In fact, the best values of NMI in the normalized influences w'_σ and w'_L are 0.145 and 0.217, respectively.

Table 5: Best and average values of NMI.

	w_C	w_σ	w_σ^λ	w_L	w_L^λ
Best	0.150	0.227	0.150	0.426	0.235
Avg.	0.150	0.165	0.127	0.266	0.156

The effects of parameters are assessed in Table 6. In the table, *FtoF* drives better results in w_L and H_{75} significantly outperforms others in w_σ and w_L. While w_σ and w_L have the same tendency on $\Delta_x^y(p)$ and α, the results on $I_x^y(p)$ is quite different between them.

Table 6: Winning rates of different models in NMI.

	w_σ	w_L		w_σ	w_L
idf	0.000	0.083	D_5	0.000	0.000
r_idf	0.000	0.333	D_{10}	0.000	0.188
tfidf	0.750	0.000	D_{20}	0.000	0.000
const	0.250	0.583	H_{75}	0.875	0.500
FtoF	0.521	0.667	H_{150}	0.125	0.250
LtoF	0.479	0.333	H_{300}	0.000	0.063

Figure 4 shows the results of comparisons on a mixture parameter λ. We can observe that small λs get better results in w_L^λ due to the supervised learning. The peak of w_σ^λ is also small relatively. These results suggests that the effect of influence is dominant than that of homophily on community discovery in this dataset.

The parameter effects are completely different from the tasks of web page recommendation and that

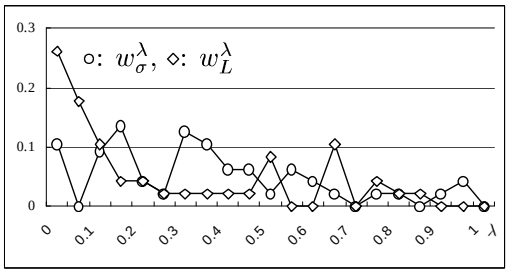

Figure 4: Winning rates of different λs in NMI.

of community discovery. Thus, we can confirm that the appropriate parameter setting heavily depends on application domain.

4 RELATED WORK

Estimation of user influence attracts much attention in the area of social network analysis, and many sophisticated models are proposed, *e.g.* (Goyal et al., 2010; Kimura et al., 2009). However, it is difficult to apply them directly to proxy logs not having precise information to construct accurate user networks.

Several methods for estimating user influence without explicit network information have been developed recently. In (Gomez Rodriguez et al., 2010), an algorithm named 'netinf' is proposed which estimates hidden network structures from a set of information cascades obtained from (proxy) log data. Netinf estimates directed unweighted networks of users by adopting the exponential waiting time model on information diffusion while it assumes that the degree of user influences are the same among any user pairs. As an extension of netinf, a convex programming based method for inferring directed weighted network structures from cascade data has been proposed in (Myers and Leskovec, 2010). While these two methods employ the exponential waiting time model for reflecting information on time difference, they do not consider the importance of contents at all.

A probabilistic model for user adoption behaviors has been proposed in (Au Yeung and Iwata, 2010). By using the model, user influence as well as influences of popularity and recency of contents are estimated from log data. The model requires a parameter specifying the length of period in which a user affects others. In other words, behaviors outside of the period are regarded to give no effect. On the other hand, the effects of behaviors decrease gradually with time in our proposal.

5 CONCLUSIONS

In this paper, we propose a framework for estimating implicit user influence from proxy logs. We model user interactions as vectors by taking account of the difference of access time and importance of web pages, and use the vectors to estimate the influence. The proposed methods are evaluated empirically by using three real datasets in the tasks of web page recommendation and community discovery.

For future work, detailed assessments of obtained user influences are necessary. In addition, we plan to investigate further experiments with large-scale proxy logs having different characteristics as well as precise comparisons with related techniques on estimating user influence.

REFERENCES

Au Yeung, C.-m. and Iwata, T. (2010). Capturing implicit user influence in online social sharing. In *Proceedings of the 21st ACM Conference on Hypertext and Hypermedia*, pages 245–254.

Chang, C.-C. and Lin, C.-J. (2001). *LIBSVM: a library for support vector machines*. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Csardi, G. and Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal*, Complex Systems:1695.

Danon, L., Díaz-Guilera, A., Duch, J., and Arenas, A. (2005). Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(9):P09008.

Gomez Rodriguez, M., Leskovec, J., and Krause, A. (2010). Inferring networks of diffusion and influence. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1019–1028.

Goyal, A., Bonchi, F., and Lakshmanan, L. V. (2010). Learning influence probabilities in social networks. In *Proceedings of the third ACM International Conference on Web Search and Data Mining*, pages 241–250.

Kimura, M., Saito, K., and Motoda, H. (2009). Efficient estimation of influence functions for sis model on social networks. In *Proceedings of the 21st International Joint Conference Artificial Intelligence*, pages 2046–2051.

McPherson, M., Lovin, L. S., and Cook, J. M. (2001). Birds of a Feather: Homophily in Social Networks. *Annual Review of Sociology*, 27(1):415–444.

Myers, S. and Leskovec, J. (2010). On the convexity of latent social network inference. In *Advances in Neural Information Processing Systems 23*, NIPS, pages 1741–1749.

Newman, M. E. J. and Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113.